

AMENDMENTS TO THE CLAIMS

Upon entry of this amendment, the following listing of claims will replace all prior versions and listings of claims in the pending application.

IN THE CLAIMS

Please amend claim 17 as follows:

1. (Previously Presented) A computer-implemented method for adaptively filtering URL messages routed across a network by generating exception rules to rejection rules based on attributes of URLs within messages previously received and rejected, the method comprising:
 - receiving a first message specifying a first URL component;
 - rejecting the first message based on a rejection rule;
 - maintaining a frequency for the first URL component, wherein the frequency is a function of a number of occurrences with which messages containing the first URL component were rejected and a number of occurrences with which messages containing descendants of the first URL component were rejected;
 - generating an exception rule for the first URL component and its descendants responsive to the frequency of the first URL component satisfying a set of constraints;
 - receiving a second message specifying the first URL component; and allowing the second message to pass.
2. (Previously Presented) The method of claim 1, wherein the set of constraints a frequency exceeding a threshold and having no children with a frequency above the threshold.

3. (Previously Presented) The method of claim 1, wherein the set of constraints requires a frequency exceeding a threshold.
4. (Original) The method of claim 1, further comprising applying the exception rule to determine whether to allow the selected URL component and its descendants.
5. (Original) The method of claim 2, wherein the threshold is a product of a total number of URL messages over a time interval and a percentage of the messages that should be allowed.
6. (Original) The method of claim 1, wherein the exception rule is generated by inferencing a scalar data type of the descendants of the selected URL component.
7. (Original) The method of claim 1, wherein the frequency is a direct count of the occurrences of the URL component.
8. (Original) The method of claim 1, wherein the frequency is a weighted count of the occurrences of the URL component.
9. (Previously Presented) A computer-implemented method for adaptively filtering URL messages routed across a network by generating exception rules to rejection rules based on attributes of URLs within messages previously received and rejected, the method comprising:

receiving a plurality of messages, each message specifying a first URL component;

rejecting the plurality of messages;

storing rejected URLs in a trie structure, wherein each node in the trie structure is associated with a URL component;

maintaining a frequency for each node associated with a URL component, wherein the frequency is a function of a number of occurrences with which a URL component associated with a node were rejected and a number of occurrences with which descendants of the URL component were rejected;

generating an exception rule for a node associated with the first URL component and its descendants responsive to the frequency of the first URL component satisfying a set of constraints;

receiving a message specifying the first URL component; and

allowing the message to pass.

10. (Original) The method of claim 9, further comprising applying the exception rule to determine whether to allow the selected node and its descendants.

11. (Previously Presented) The method of claim 9, wherein the set of constraints requires a number of occurrences exceeding a threshold.

12. (Previously Presented) The method of claim 9, wherein the set of constraints requires a number of occurrences exceeding a threshold and having no children with a number of occurrences above the threshold.

13. (Original) The method of claim 11, wherein the threshold is a product of a total number of URL messages over a time interval and a percentage of the messages that should be allowed to pass.

14. (Original) The method of claim 9, wherein the exception rule is generated by inferencing a scalar data type of the descendants of the selected URL component.

15. (Original) The method of claim 9, wherein the frequency is a direct count of a number of occurrences of the URL component.

16. (Original) The method of claim 9, wherein the frequency is a weighted count of a number of occurrences of the URL component.

17. (Previously Presented) A system for adaptively filtering URL messages routed across a network, by generating exception rules to rejection rules based on attributes of URLs within messages previously received and rejected, the system comprising:

a learning engine ~~adapted to perform the steps of:~~

storing rejected URLs in a trie structure, wherein each node in the trie structure is associated with a URL component;

maintaining a frequency for each node associated with a URL component,

wherein the frequency is a function of a number of occurrences with which a URL component associated with a node were rejected and a number of occurrences with which descendants of the URL component were rejected, and

generating an exception rule for a first node and its descendants, responsive to the frequency of the URL component associated with the first node satisfying a set of constraints; and

a filter ~~configured to applying~~ configured to apply the exception rule to determine whether to allow the first node and its descendants.

18. (Canceled)

19. (Previously Presented) The system of claim 17, wherein the set of constraints requires a number of occurrences exceeding a threshold and having no children with a number of occurrences above the threshold.

20. (Previously Presented) The system of claim 17, wherein the set of constraints requires a frequency exceeding a threshold.

21. (Previously Presented) The system of claim 19, wherein the threshold is a product of a total number of URL messages over a time interval and a percentage of the messages that should be allowed.

22. (Original) The system of claim 17, wherein the exception rule is generated by inferencing a scalar data type of the descendants of the selected node.

23. (Previously Presented) The system of claim 17, wherein the frequency is a direct count of the number of occurrences of the URL component.

24. (Previously Presented) The system of claim 17, wherein the frequency is a weighted count of the number of occurrences of the URL component.

25. (Previously Presented) A computer program product comprising: a computer-readable medium having computer program code embodied therein for adaptively filtering URL messages routed across a network by generating exception rules to rejection rules based on attributes of URLs within messages previously received and rejected, the computer program code adapted to: store rejected URLs in a trie structure, wherein each node in the trie structure is associated with a URL component and each node associated with a URL component maintains a frequency, wherein the frequency is a function of a number of occurrences with which a URL component associated with a node were rejected; and generate an exception rule for a first node and its descendants responsive to the frequency of the a URL component associated with the first node satisfying a set of constraints.

26. (Previously Presented) The computer program product of claim 25, wherein each node associated with a URL component maintains a frequency, wherein the frequency is a function of a number of occurrences with which a URL component associated with a node was rejected and a number of occurrences with which descendants of the URL component were rejected.

27. (Previously Presented) The computer program product of claim 26, wherein the set of constraints requires a frequency exceeding a threshold and having no children with a frequency above the threshold.

28. (Previously Presented) The computer program product of claim 26, wherein the set of constraints requires a frequency exceeding a threshold.

29. (Original) The computer program product of claim 25, wherein the computer program code is further adapted to apply the exception rule to determine whether to allow the selected node and its descendants to pass.

30. (Original) The computer program product of claim 27, wherein the threshold is a product of a total number of URL messages over a time interval and a percentage of the messages that should be allowed to pass.

31. (Original) The computer program product of claim 25, wherein the exception rule is generated by inferencing a scalar data type of the descendants of the selected node.

32. (Previously Presented) A computer-implemented method for adaptively filtering URL messages routed across a network, by generating exception rules to rejection rules based on attributes of URLs within messages previously received and rejected, the method comprising:

storing rejected URLs in a trie structure, wherein each node in the trie structure is associated with a URL component; and each node associated with a URL component maintains a

frequency, wherein the frequency is a function of a number of occurrences with which a URL component associated with a node was rejected; and

and generating an exception rule for a first node and its descendants, responsive to the frequency of the URL component associated with the first node satisfying a set of constraints.

33. (Original) The method of claim 32, further comprising maintaining a frequency for each node associated with a URL component, wherein the frequency is a function of a number of occurrences with which a URL component associated with a node was rejected and a number of occurrences with which descendants of the URL component were rejected.

34. (Original) The method of claim 32, further comprising applying the exception rule to determine whether to allow the selected node and its descendants.

35. (Previously Presented) The method of claim 32, wherein the set of constraints requires a number of occurrences exceeding a threshold.

36. (Previously Presented) The method of claim 32, wherein the set of constraints requires a number of occurrences exceeding a threshold and having no children with a number of occurrences above the threshold.

37. (Original) The method of claim 35, wherein the threshold is a product of a total number of URL messages over a time interval and a percentage of the messages that should be allowed.

38. (Original) The method of claim 32, wherein the exception rule is generated by inferencing a scalar data type of the descendants of the selected URL component.

39. (Original) The method of claim 33, wherein the frequency is a direct count of the number of occurrences of the URL component associated with the selected node.

40. (Original) The method of claim 33, wherein the frequency is a weighted count of the number of occurrences of the URL component associated with the selected node.